A New Linguistic Prediction Method Based on Random Set Semantics

N. J. Randon and **J. Lawry**
A.I. Group, Department of Engineering Mathematics,
University of Bristol BS8 1TR, United Kingdom,
{Nick.Randon, J.Lawry}@bris.ac.uk

Abstract

In this paper we propose a random set framework for learning linguistic models for prediction problems. In this framework we show how we can model prediction problems based on learning linguistic prototypes defined using joint mass assignments on sets of labels. The potential of this approach is then demonstrated by its application to a model and by benchmark problem and comparing the results obtained with those from other state-of-the-art learning algorithms. We then show how this framework can be used to evaluate linguistic hypothesises using the learnt prototype models.

1 Introduction

The idea of using fuzzy sets to represent words was first proposed by Zadeh [16], who stated that fuzzy memberships could be used to model the imprecision and ambiguity of natural language terms such as *small*, *medium* and *large*. However, this generates a number of problems in terms of semantics and computational complexity (see [9] for a discussion).

Here we propose an alternative framework introduced by Lawry (see [10]). This approach uses fuzzy sets to partition an attributes domain into linguistic labels. Random sets (see [3]) and mass assignments are then used as a method for evaluating the appropriateness of the labels for a given value. Prediction is carried out using mass assignment prototypes representing relationships between input and output attributes at the label level. These prototypes are obtained by aggregating linguistic descriptions of examples on the prediction space from a database. The models are then used in conjunction with a Naïve or Semi-Naïve-Bayes classifier (see [8] and [11]) together with a defuzzification method to perform prediction.

2 Label Semantics

Suppose we have an attribute x with domain Ω and we ask a set of experts V to provide a finite set of labels LA with which to describe x. For $x \in \Omega$ we ask each of the experts E to supply us with a subset of LA that they deem as appropriate to describe x. This generates a set of labels describing x denoted $\mathcal{D}_x^E$. As each of the experts is likely to have a different subset of appropriate labels to describe the situation, we obtain a random set $\mathcal{D}_x$ across the power set of LA as we vary between experts. By combining the label description provided by the experts we can determine a mass assignment on the power sets of LA (2^{LA}) representing the distribution of the random set $\mathcal{D}_x$.

Definition 1 (Mass Assignment) *A mass assignment on 2^Ω is a function $m : 2^\Omega \rightarrow [0, 1]$ such that:*

$$\sum_{S \subseteq \Omega} m(S) = 1$$

Definition 2 (Value Description) *Let V be the set of experts. For $x \in \Omega$ the label description of x is a random set from V into the power set of LA, denoted $\mathcal{D}_x$, with associated mass assignment m_x:*

$$\forall S \subseteq LA \; m_x(S) = P_V(\{E \in V : \mathcal{D}_x^E = S\})$$

where P_V is the prior probability distribution over the population V.

For any mass assignment on 2^{LA} it is likely to be the case that only a subset of 2^{LA} will have non-zero mass. These sets are referred to as focal sets of LA.

Definition 3 (Focal Sets) *The focal sets for the labels LA are defined as the union of the focal sets for the mass assignment m_x as x varies across Ω.*

$$\mathcal{F}_{LA} = \{S \subseteq LA | \exists x \in \Omega, m_x(S) > 0\}$$

We can formally define a measure for the appropriateness of a label L for a value x, denoted $\mu_L(x)$, by evaluating the mass of those label sets containing L.

Definition 4 (Appropriateness Degrees)

$$\forall x \in \Omega, \forall L \in LA \ \mu_L(x) = \sum_{S \subseteq LA : L \in S} m_x(S)$$

Notice that $\mu_L : \Omega \rightarrow [0,1]$ and hence, corresponds to a fuzzy set on Ω. However, the term fuzzy set does not seem entirely suitable in this context since we are not measuring a degree of membership but rather a degree of appropriateness.

Here we have assumed that we have knowledge of the underling expert behaviour but in many situations this is not the case. Hence, we need to define a mapping from appropriateness degrees to mass assignments. To achieve this we make the assumption that individuals in V differ regarding what labels are appropriate for a value only in terms of generality and specificity. This is referred to as the consonance assumption. Also, we make the further assumption $\forall x \in \Omega \ \underset{L \in LA}{max} \ \mu_L(x) = 1$.

Definition 5 (Consonance Mapping) *Let* $\{\mu_L(x) : L \in LA\} = \{y_1, \ldots, y_n\}$ *be ordered such that* $y_i > y_{i+1}$ *for* $i = 1, \ldots, n-1$ *then for* $S_i = \{L \in LA : \mu_L(x) \geq y_i\}$,

$$m_x(S_i) = y_i - y_{i+1} \text{ for } i = 1, \ldots, n-1$$
$$m_x(S_n) = y_n, \ m_x(\emptyset) = 1 - y_1$$

Example 1 *Suppose we have an attribute on the domain [0, 114] with associated labels* $LA = \{$*very small(vs), small(s), medium(m), large(l), very large(vl)*$\}$, *defined according to the following trapezoidal fuzzy sets:*

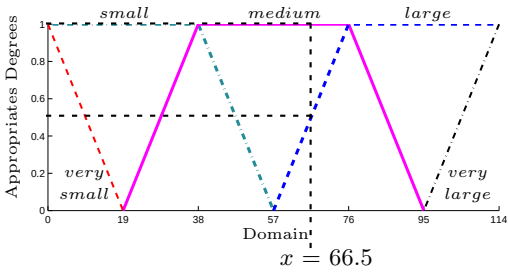


Figure 1: Appropriates degrees for $LA = \{vs, s, m, l, vl\}$

For $x = 66.5$ *we have* $\mu_m(x) = 1$ *and* $\mu_l(x) = 0.5$. *From this it is possible to construct the consonant mass assignment for the point* x *as follows:*

$m_{66.5} = \{medium\} : 0.5, \ \{medium, \ large\} : 0.5$

If x *is now allowed to vary across the domain [0, 114] we obtain a functional definition for* m_x *as shown in figure 2.*

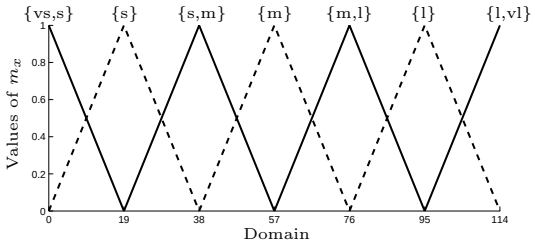


Figure 2: Mass assignment descriptions for x

Clearly, the framework described in this section is related to the random set semantics for fuzzy memberships proposed by Goodman and Nguyen [4]. However, the latter defines random sets on subsets of the attribute universe while for the current framework they are defined on subsets of labels. This provides an interesting new perspective and allows for a more straightforward treatment of continuous domains.

3 Label Prototypes for Modelling Prediction Problems

Consider a prediction problem where the objective is to model the relationship between input attributes $x_1, \ldots, x_{n-1}$ and output attribute x_n. Label sets LA_j are defined on input universes $\Omega_j : \ j = 1 \ldots, n-1$ and a set of labels LC is also defined on the output universe Ω_n. Each $L \in LC$ is represented by a trapezoidal fuzzy set on the prediction space. The focal sets of LC are given by $\mathcal{F}_{LC} = \{S \subseteq LC | \exists x_n \in \Omega_n, m_{x_n}(S) > 0\} = \{F_j\}_j$.

Suppose we have a training set of examples $DB = \{\langle x_1(i), \ldots, x_n(i) \rangle | i = 1, \ldots, N\}$. The input attributes $x_1, \ldots, x_{n-1}$ are now partitioned into subsets $S_1, \ldots, S_w$ where $w \leq n-1$ and for each $F_j \in \mathcal{F}_{LC}$ a joint mass assignment $m_{i,j}$ is determined as follows: Suppose, w.l.o.g. that $S_i = \{x_1, \ldots, x_v\}$ then the joint mass assignment on $2^{LA_1} \times \cdots \times 2^{LA_v}$ conditional on $F_j \in \mathcal{F}_{LC}$ is defined by: $\forall T_r \in 2^{LA_r} : r = 1, \ldots, v \ \forall F_j \in \mathcal{F}_{LC}$

$$m_{i,j}(T_1, \ldots, T_v) = \frac{\sum_{k \in DB} m_{x_n(k)}(F_j) \prod_{r=1}^{v} m_{x_r(k)}(T_r)}{\sum_{k \in DB} m_{x_n(k)}(F_j)}$$

Hence, the prototype describing F_j is the vector: $\langle m_{1,j}, \ldots, m_{w,j} \rangle$.

4 Prediction Using Prototypes on Linguistic Class

We now give details of how prediction can be performed using linguistic class prototypes together with a Semi-Naïve-Bayes (see [8]) learning algorithm. We use Semi-Naïve-Bayes in this context to weaken the independence assumption of Naïve-Bayes (see [11]). This is achieved by defining joint mass assignments to model dependences between attributes in variable groupings and then assuming independence between groupings. We then carry out a defuzzification step to obtain a prediction value from this model.

Bayes theorem is used here to evaluate the probability of each of the focal elements F_j given a vector of input values $\langle x_1, \ldots, x_{n-1} \rangle$ as follows:

$$Pr(F_j|x_1, \ldots, x_{n-1}) = \frac{Pr(F_j) \prod_{r=1}^{w} p(S_r|F_j)}{\sum_k Pr(F_k) \prod_{r=1}^{w} p(S_r|F_k)}$$

Where $Pr(F_j) = \frac{1}{|DB|} \sum_{k \in DB} m_{x_n(k)}(F_j)$. There is now the problem of how to estimate the density function $p(x_1, \ldots, x_{n-1} \mid F_j)$. Consider the joint mass assignment for grouping S_i given F_j. If we assume that there is a uniform prior distribution on $\times_{r=1}^{v} \Omega_r$ then the joint prior mass assignment on $\times_{r=1}^{v} 2^{LA_r}$ is: $\forall T_i \subseteq LA_i : i = 1, \ldots, v$

$$pm(T_1, \ldots, T_v) = \prod_{i=1}^{v} \int_{\Omega_i} m_{x_i}(T_i) u_i(x_i) dx_i$$

Where $u(x_1, \ldots, x_v) = \prod_{i=1}^{v} u_i(x_i)$ is the uniform distribution on $\times_{r=1}^{v} \Omega_r$ and $u_r(x_r)$ the uniform distribution on Ω_r. From this we can define the joint density on $x_1, \ldots, x_v$ conditional on $m_{i,j}$:

$$p(S_i|m_{i,j}) = p(x_1, \ldots, x_v|m_{i,j}) = u(x_1, \ldots, x_v) \sum_{T_1 \times \cdots \times T_v} \frac{m_{i,j}(T_1, \ldots, T_v)}{pm(T_1, \ldots, T_v)} \prod_{r=1}^{v} m_{x_r}(T_r)$$

This calculation is motivated by the theorem of total probability (see [12]) which for a one dimensional mass assignment, describing variable x on Ω, is as follows: $\forall a \in \Omega$

$$\begin{aligned} p(a|m) &= \sum_{S \subseteq LA} p(a|\mathcal{D}_x = S) Pr(\mathcal{D}_x = S) \\ &= \sum_{S \subseteq LA} p(a|\mathcal{D}_x = S) m(S) \end{aligned}$$

$$\begin{aligned} also: \; p(a|\mathcal{D}_x = S) &= \frac{Pr(\mathcal{D}_x = S|x = a) u(a)}{Pr(\mathcal{D}_x = S)} \\ &= \frac{m_a(S) u(a)}{pm(S)} \end{aligned}$$

Hence, making the relative substitution and simplifying we obtain the expression:

$$p(a \mid m) = u(a) \sum_{S \subseteq LA} \frac{m(S)}{pm(S)} m_a(S)$$

By taking $p(S_i| \; F_j) \cong p(S_i| \; m_{i,j})$ for each grouping S_i, we now have the following Semi-Naïve-Bayes calculation:

$$Pr(F_j|x_1, \ldots, x_{n-1}) \propto \frac{Pr(F_j) \prod_{r=1}^{w} p(S_r|m_{r,j})}{\sum_k Pr(F_k) \prod_{r=1}^{w} p(S_r|m_{r,k})}$$

We now define a defuzzification method to determine the predicted value for x_n as follows: Assuming there is a uniform prior distribution on $x_1, \ldots, x_{n-1}$, then, by evaluating $Pr(F_j|x_1, \ldots, x_{n-1})$ for all F_j we obtain a mass assignment on $\mathcal{F}_{LC}$. This can then be mapped to a distribution on x_n as follows:

$$p(x_n|x_1, \ldots, x_{n-1}) = \sum_j Pr(F_j|x_1, \ldots, x_{n-1}) p(x_n|F_j)$$

$$where: \; p(x_n|F_j) = \frac{m_{x_n}(F_j)}{\int_{\Omega_n} m_{x_n}(F_j) \; dx_n}$$

We then take our estimate of x_n, denoted $\widehat{x}_n$, to be the expected value of the distribution:

$$\begin{aligned} \widehat{x}_n &= \int_{\Omega_n} x_n \; p(x_n|x_1, \ldots, x_{n-1}) \; dx_n \\ &= \sum_j \; Pr(F_j|x_1, \ldots, x_{n-1}) \; E(x_n|F_j) \end{aligned}$$

An alternative defuzzification method is obtained by replacing $E(x_n|F_j)$ by the mode of the distribution $p(x_n|F_j)$ (i.e. $argmax(m_{x_n}(F_j))$).

5 Grouping Methods

In this section we introduce a number of methods for automatically finding attribute groupings that increase discrimination in the model. In general it is too computationally expensive to search the complete problem space of all attribute groupings and then partition to see if discrimination can be increased, as the search space would be exponential. To counter this problem a heuristic search has been proposed, based on

the order of importance of each of the attribute groupings S_i. The heuristic used to estimate the importance is defined as follows:

Definition 6 (Importance Measure) *Let the joint mass assignment for S_i given F_j be denoted $m_{i,j}$. For any input vector S_i the probability of the focal set F_j can be estimated using Bayes theorem:*

$$IM_j(S_i) = \frac{\sum_{k \in DB} Pr(F_j|S_i(k))\ m_{x_n}(F_j)}{\sum_{k \in DB} Pr(F_j \mid S_i(k))}$$

$$where: Pr(F_j|S_i) = \frac{p(S_i|m_{i,j})Pr(F_j)}{p(S_i|m_{i,j})Pr(F_j) + p(S_i|m_{i,\neg j})(1 - Pr(F_j))}$$

where $m_{i,\neg j}$ is the mass assignment for group S_i conditional on $\mathcal{F}_{LC} - \{F_j\}$

$IM_j(S_i)$ is a measure of importance of the set of variables S_i as discriminators of F_j from the other focal sets. The closer $IM_j(S_i)$ is to 1 the more discriminating the group S_i. In this case $\sum_{k \in DB} Pr(F_j|S_i(k)) m_{x_n(k)}(F_j)$ is high relative to $\sum_{k \in DB} Pr(F_j|S_i(k)) \left(1 - m_{x_n(k)}(F_j)\right)$.

Due to the '*curse of dimensionality*' (see [2]) careful limits must be set on the maximum number of attributes that can be grouped when running this algorithm. The effect of which can be limited by trading granularity off against dimensionality. The importance measure here is now combined with two search strategies to find discriminative groupings:

5.1 Guided Breadth First Search

Consider a breadth first search where the most important current grouping S_i is combined with all the other current groupings to see if the combination significantly increases discrimination. Next the second most important grouping is tested with the remaining unused groupings and so on. At the next stage the new groupings produced are tested in a similar manner and this continues until a terminating condition is satisfied. This method provides a fairly extensive search of the space of the partitions, but does limit the structure of the groupings generated.

5.2 Guided Depth First Search

Alternatively, consider a depth first search where the most important grouping S_i is tested with all other groupings to see if the combination increases discrimination. Next any new grouping produced is tested with the unused groupings to see if discrimination is further increased. This continues until some termination condition is satisfied. The process is repeated with the next most important unused grouping and so on, until all unused grouping have been tested. This allows for a richer structure of groupings but has the disadvantage that some important groupings may be missed.

We now define two methods for measuring whether or not a pair of attributes should be combined. The first is based on a direct measure of correlation and the second is based on a measure of the change in importance resulting from grouping.

Definition 7 (Correlation Measure) *Let $\mathcal{F}_1$ be the focal sets for S_1 and $\mathcal{F}_2$ the focal sets for S_2. Now let $m_{1,2,j}$ be the joint mass of $S_1 \cup S_2$ given the output focal set F_j.*

$$CORR(S_1, S_2) = \sqrt{\frac{1}{|\mathcal{F}_1||\mathcal{F}_2|} \sum_{R \subseteq \mathcal{F}_1} \sum_{T \subseteq \mathcal{F}_2} (m_{1,2,j}(R,T) - m_{1,j}(R) m_{2,j}(T))^2}$$

Here a threshold is used so that the nearer the correlation measure is to 1, the more likely it is that grouping will take place. An alternative to measuring correlation is to trying to maximise the increase in importance of any new grouping formed.

Definition 8 (Improvement Measure) *Suppose we have two subsets of attributes S_1 and S_2 then the improvement in importance obtained by combining them can be calculated as follows:*

$$IPM_j(S_1, S_2) = \frac{\min(IM_j(S_1), IM_j(S_2))}{IM_j(S_1, S_2)}$$

A threshold is once again used so that the closer the improvement measure is to 0 the more likely that the attributes will be combined.

6 Performance on a Benchmark Problem

We now give details of the performance of the proposed prediction system. The results obtained from the Fuzzy Bayesian methods are compared here to a ε-Support Vector Regression system (ε-SVR) [14], implemented in [6] by Gunn [5]. The ε-SVR was implemented using a gaussion Radial Basis Function (RBF) kernel with an ε-insensitive loss function.

We now define a method for evaluating the prediction error, the Mean Square Error (MSE), which is calculated as follows:

$$MSE = \frac{1}{|DB|} \sum_{i \in DB} (\widehat{x}_n(i) - x_n(i))^2$$

6.1 Surface Based on: $z = \sin(x \times y)$

In this example a training set of 529 points were generated describing a surface defined according to the equation $z = \sin(x \times y)$ where $x, y \in [0, 3]$, as shown in figure 3:

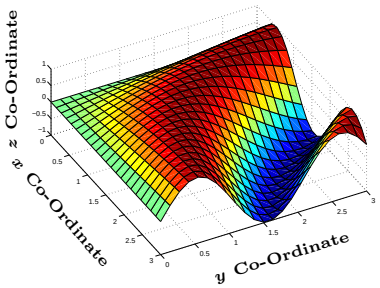
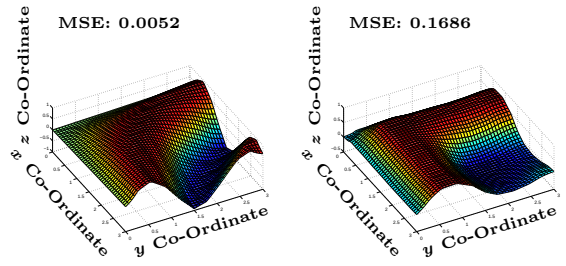


Figure 3: Surface defined by the 529 points.

The prototype models were generated from 7 labels being defined over the three attributes domains x, y and z. The fuzzy labels were defined by using a percentile method to obtain a crisp partition with an equal number of data points falling within each crisp set and then projecting trapezoidal fuzzy sets over this partition. As there are only two input attributes the choice of search method is arbitrary, as both will obtain the same results. For the correlation method a threshold of 0.005 was used and for the improvement measure a threshold of 0.895 was used.

From training the system over the 529 points, and testing on a denser grid of 2,209 points the following predictions for both the correlation and improvement measure were obtained (see figure 4(a)). Figure 4(a) can be directly compared to the surface obtained by applying Fuzzy Naïve-Bayes (see figure 4(b)). From this it can be seen that the prediction accuracy is significantly increased by using the Semi-Naïve-Bayes approach.



(a) Semi-Naïve-Bayes Prediction.

(b) Naïve-Bayes Prediction.

Figure 4: Prediction surfaces obtained for both Naïve and Semi-Naïve-Bayes.

We now can compare these results to those obtained by applying the ε-SVR to the same data set and setting the parameters as follows: $\sigma = 1$, $\varepsilon = 0.05$, $C = \infty$. From this it was found that the ε-SVR method obtains a marginally better prediction of the surface with an MSE of 0.0011 which is an improvement of 0.0041 compared to that obtained using Semi-Naïve-Bayes. Though this difference may be reduced by using more labels to describe the attributes.

6.2 Prediction of Sunspots

This problem is from the time series data library [7] and contains data on J.R. Wolf and Zürich sunspot relative numbers [1] between the years 1700-1979. The data was organized as described in [15], except that the validation set of 35 examples (1921-1955) was merged into the test set of 24 examples (1956-1979). This is because a validation set is not required in the fuzzy label framework. Hence, a training set of 209 examples (1712-1920) and a test set of 59 examples (1921-1979) were used. The input attributes were x_{t-12} to x_{t-1} and the output attribute was x_t. Each attribute had 4 labels defined over the domains using a percentile method to obtain the fuzzy partition. The correlation threshold was set to 0.005 and the improvement threshold set to 0.895, with a maximum allowed grouping size of 7 attributes. Figure 5 gives details of the prediction results obtained:

	MSE	
	Training	Test
Naïve-Bayes	493.914	810.742
Depth first search:		
Correlation Measure	290.325	506.6
Improvement Measure	134.704	499.659
Breadth first search:		
Correlation Measure	376.136	539.571
Improvement Measure	219.864	615.07

Figure 5: Prediction result obtained for the sunspot data set showing the MSE

Figure 5 shows that the depth first search using the improvement measure obtains the best result, with a significant increase over Naïve-Bayes. Some caution must be taken in interpreting these results as the thresholds used are not optimised, hence, for a different threshold value it is possible that the correlation measure would obtained the same prediction results as the improvement measure.

The result obtained here from applying the fuzzy prediction method can again be directly compared to those obtained by applying the ε-SVR to the problem. Here the parameters of

the ε-SVR were set as follows: $\sigma = 3$, $\varepsilon = 0.05$, $C = 5$. Form this the results shown in figure 6 were obtained. Figure 6 shows that the ε-SVR obtained a similar but slightly better prediction result, however, we must be careful in drawing conclusions, as we are comparing un-optimised result for both system.

Test set results	**MSE**
ε-Support Vector Regression system	418.126
Best Semi-Naïve-Bayes	499.659

Figure 6: Prediction results obtained for the sunspot prediction test set from applying ε-SVR with the following parameters: $\sigma = 3$, $\varepsilon = 0.05$, $C = 5$, and the best Semi-Naïve-Bayes prediction.

We can further compare our results to those given in [15] using the suggested measure of prediction accuracy, Average Relative Variance (AVR), which is calculated as follows:

$$ARV(DB) = \frac{1}{\widehat{\sigma}^2}\frac{1}{N}\sum_{k \in DB}(x_k - \widehat{x}_k)^2$$

Figure 7 show the results obtained using our best Semi-Naïve-Bayes method (a depth first search with the improvement measure) and the ε-SVR are better to those stated by Weigend *et al.* [15] with Semi-Naïve-Bayes performing the best on the 1956-1979 segment of the test set. It should be highlighted that the results of Weigend *et al.* [15] for the years 1712-1920 and 1921-1955 are significantly better. This is because these time periods corresponded to the training and validation sets used to train the neural network. The disparity between the results seen for the years 1712-1920 and 1921-1955 and those stated by Weigend *et al.* over the 1956-1979, suggest over-fitting by the network. However, for a full and fair comparison of the results here we must also allow the validation set to be included in the training sets for the fuzzy Bayesian approach. This is because the validation set is used during the training process. The results in this case are given in the bottom two rows of figure 7. This shows that, as we would be expected, we obtain better prediction results for both Semi-Naïve-Bayes and the ε-SVR system on the validation set (1921-1955) which now more closely matches the results given by Weigend *et al.*. Also we see that in this instance there is little different in the prediction obtained by using Semi-Naïve-Bayes and by using the ε-SVR system. Further we can give a direct comparison between the predicted results from both the ε-SVR and best Semi-Naïve-Bayes prediction result. (see figuer 8).

Average Relative Variance			
	1712 1920	1921 1955	1956 1979
Single step prediction: (see [15] p 414)			
Weight Elimination Net.	0.082	0.086	0.35
TRA Model	0.097	0.097	0.28
Results from merging the validation with test			
Best Semi-Naïve-Bayes	0.113	0.204	0.254
ε-SVRsystem	0.133	0.117	0.263
Results from merging the validation with training			
Best Semi-Naïve-Bayes	0.135	0.108	0.249
ε-SVR system	0.127	0.087	0.248

Figure 7: Full comparison of results with those obtained by *Weigend et al.* [15].

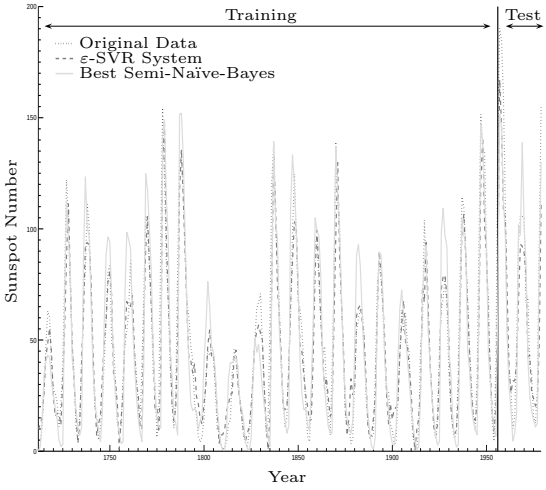


Figure 8: Comparison of prediction result obtained form ε-SVR and best Semi-Naïve-Bayes method.

7 Query Evaluation

Often we what more form a model than just the ability to obtain a prediction. In addition we want to use the model to infer relationships and to test hypothesises. We now propose a methodology for evaluating linguistic queries within the prototype framework. Here the queries are restricted to those that can be expressed in the form of a vector, $\vec{\theta} = \langle \theta_1, \ldots, \theta_n \rangle$, where θ_i is a label expression generated by recursive application of the logical connectives to labels in LA_i. Stated in terms of Zadeh's linguistic constraints this vector represents the expression x_1 *is* θ_1 *and* x_2 *is* θ_2 *and* $\ldots$ *and* x_n *is* θ_n.

In the query mechanism we need to be able to evaluate compound label expression. For example, we may wish to know whether or not expressions such as $medium \wedge low$, $medium \vee low$ and $\neg high$ are appropriate to describe a value $x \in \Omega$. In the context of this framework we interpret the main logical connectives in the following manner: $L_1 \wedge L_2$ means that both L_1 and L_2 are appropriate labels, $L_1 \vee L_2$ means

that either L_1 or L_2 are appropriate labels and $\neg L$ means that L is not an appropriate label. More generally, if we consider label expressions formed from LA by recursive application of the connectives then an expression θ identifies a set of possible label sets $\lambda(\theta)$ as follows:

Definition 9 (Label Expression) *The set of label expression of LA, denoted LE, are defined recursively as follows:*

(i) $L_i \in LE$ *for* $i = 1, \ldots, n$

(ii) *if* $\theta, \varphi \in LE$ *then* $\neg\theta,\ \theta \vee \varphi,\ \theta \wedge \varphi$

Definition 10 *For* $L \in LA$ $\lambda(L) = \{S \subseteq LA : L \in S\}$ *and for label expressions* θ *and* φ.

(i) $\forall L_i \in LA\ \lambda(L_i) = \{S \subseteq LA \mid L_i \in S\}$

(ii) $\lambda(\theta \wedge \varphi) = \lambda(\theta) \cap \lambda(\varphi)$

(iii) $\lambda(\theta \vee \varphi) = \lambda(\theta) \cup \lambda(\varphi)$

(iv) $\lambda(\neg\theta) = \overline{\lambda(\theta)}$.

Intuitively, $\lambda(\theta)$ corresponds to those subsets of LA identified as being possible values of $\mathcal{D}_x$ by expression θ. In this sense the imprecise linguistic restriction 'x *is* θ' on x corresponds to the strict constraint $\mathcal{D}_x \in \lambda(\theta)$ on $\mathcal{D}_x$. The notion of appropriateness measure given above can now be extended so that it applies to compound label expressions. The idea here is that $\mu_\theta(x)$ quantifies the degree to which expression θ is appropriate to describe x.

$$\mu_\theta(x) = \sum_{S \in \lambda(\theta)} m_{\mathcal{D}_x}(S)$$

We specify an output expression on the focal sets that represent the class labels LC. Hence, we can now define three queries on $F_j \in LC$, as follows:

Type I Queries: $\langle \boldsymbol{\theta_1}, \ldots, \boldsymbol{\theta_{n-1}} \rangle : \boldsymbol{F_j}$

This represents the question: *Do elements of* F_j *satisfy* $\vec{\theta}$ *?*

$$Pr(\vec{\theta}|F_j) = \sum_{T_1 \in \lambda(\vec{\theta}_1)} \cdots \sum_{T_{n-1} \in \lambda(\vec{\theta}_{n-1})} \prod_{r=1}^{v} m_{r_j}(T_i : x_i \in S_r)$$

where the description of x_n is F_j.

This value can be viewed as quantifying the appropriateness of the vector $\vec{\theta}$ to describe elements in DB for which the description of x_n is F_j (i.e. $\mathcal{D}_x = F_j$) and is denoted $\mu_{\vec{\theta}}(F_j)$.

Type II Queries: $\langle \boldsymbol{\theta_1}, \ldots, \boldsymbol{\theta_{n-1}} \rangle$

This represents the question: *Do elements of* DB *satisfy* $\vec{\theta}$ *?*

$$Pr(\vec{\theta}) = \sum_{k=1}^{j} Pr(F_k)\mu_{\vec{\theta}}(F_k)$$

This value can be viewed as quantifying the appropriateness of the vector $\vec{\theta}$ to describe the whole database DB and is denoted $\mu_{\vec{\theta}}(DB)$

Type III Queries: $\boldsymbol{F_j} : \langle \boldsymbol{\theta_1}, \ldots, \boldsymbol{\theta_{n-1}} \rangle$

This represents the question: *Do elements satisfying* $\vec{\theta}$ *have a value of* x_n *with description* F_j *?*

$$Pr(F_j|\vec{\theta}) = \frac{Pr(\vec{\theta}|F_j)Pr(F_j)}{\mu_{\vec{\theta}}(DB)}$$

Here we can derive the Type I query using the Type II and III queries, as follows:

$$Pr(\vec{\theta}|F_j) = \frac{Pr(F_j|\vec{\theta})\mu_{\vec{\theta}}(DB)}{Pr(F_j)}$$

Example 2 *Consider the* $z = \sin(x \times y)$ *problem where we defined 7 labels on the x, y and z universes, as follows:*

$$LA_x = LA_y = LC_z = \{extremely\ small(es),\ very\ Small(vs),\ small(s),\ medium(m),\ large(l),\ very\ large(vl),\ extremely\ large(el)\}$$

From this we obtain the focal elements describing the attributes universes:

$$\mathcal{F}_x = \mathcal{F}_y = \mathcal{F}_z = \{\{es, vs\}, \{vs\}, \{vs, s\}, \{s\}, \{s, m\}, \{m\}, \{m, l\}, \{l\}, \{l, vl\}, \{vl\}, \{vl, el\}\}$$

If we assume we have a fully composed model we have 11 join mass assignments on $2^{LA_x} \times 2^{LA_y}$ *conditional on* $F_j \in \mathcal{F}_z$. *Then the prototypes for each focal element of* LC_z *are described as the one dimensional vectors* $\langle m_{\{F_j\}} \rangle$. *We can now consider an example query:*

Type I: *What is the probability that the x coordinate is large, but not medium and the y coordinate is either medium and not small, or very large but not extremely large, given the focal set* $\{vs, s\}$ *?*

This query has the following vector form:

$$\langle large \wedge \neg medium, (medium \wedge \neg small) \vee (very\ large \wedge \neg extremely\ large) \rangle : \{vs, s\}$$

Specifically we have to evaluate:

$$\mu_{\vec{\theta}}(F_{\{vs,s\}}) = \sum_{T_x \in \lambda(\vec{\theta}_x)} \sum_{T_y \in \lambda(\vec{\theta}_y)} m_{\{vs,s\}}(T_x, T_y)$$

Here it is sufficient to sum $m_{\{vs,s\}}$ over $(\lambda(\theta_x) \cap \mathcal{F}_x) \times (\lambda(\theta_y) \cap \mathcal{F}_y)$ since all other relevant cells have zero mass. In this case:

$$\lambda(l \wedge \neg m) \cap \mathcal{F}_x = \{\{l\}, \{l, vl\}\}$$
$$\lambda(vl \wedge \neg h)) \cap \mathcal{F}_y = \{\{vl\}, \{vl, l\}, \{l, m\}, \{m\}\}$$

Hence, the required value is given by:

$$\begin{aligned}
&\mu_{\vec{\theta}}(F_{\{vs,s\}}) = \\
&m_{\{vs,s\}}(\{l\}, \{vl\}) + m_{\{vs,s\}}(\{l\}, \{vl, l\}) \\
&+ m_{\{vs,s\}}(\{l\}, \{l, m\}) + m_{\{vs,s\}}(\{l\}, \{m\}) \\
&+ m_{\{vs,s\}}(\{l, vl\}, \{vl\}) + m_{\{vs,s\}}(\{l, vl\}, \{vl, l\}) \\
&+ m_{\{vs,s\}}(\{l, vl\}, \{l, m\}) + m_{\{vs,s\}}(\{l, vl\}, \{m\}) \\
&= 0.0396 + 0.0088 + 0.0519 + 0.0557 \\
&+ 0.0163 + 0.0601 + 0.0019 + 0.059 \\
&= 0.2934
\end{aligned}$$

We can extent the Type I and Type III queries so that we can perform queries on compound expressions generated from LC, denoted Δ, rather than on the focal sets of LC. Here we can no longer can compute the Type I query directly. Instead we must calculate this using a Bayesian argument on the type III query as follows:

Compound Type III: $\Delta : \langle \theta_1, \ldots, \theta_{n-1} \rangle$

This represents the question: *Do elements satisfying $\vec{\theta}$ also satisfy x_n is Δ ?*

$$Pr(\Delta|\vec{\theta}) = \sum_{F_j \in \lambda(\Delta)} Pr(F_j|\vec{\theta})$$

Compound Type I: $\langle \theta_1, \ldots, \theta_{n-1} \rangle : \Delta$

This represents the question: *Do elements satisfying x_n is Δ also satisfy $\vec{\theta}$?*

$$Pr(\vec{\theta}|\Delta) = \frac{Pr(\Delta|\vec{\theta})\mu_{\vec{\theta}}(DB)}{Pr(\Delta)}$$

$$where: \; Pr(\Delta) = \sum_{\mathcal{F}_j \in \lambda(\Delta)} Pr(\mathcal{F}_j)$$

This value can be viewed as quantifying the appropriateness of the vector $\vec{\theta}$ to describe elements in DB for which the description of x_n is Δ and is subsequently denoted as $\mu_{\vec{\theta}}(\Delta)$.

8 Conclusion

We have introduced a framework for modelling fuzzy labels and shown how this can be applied to the induction of fuzzy models for prediction. In this context input-output relationships are represented by prototypes comprised of vectors of mass assignments. Each of the mass assignments is defined over the label sets describing some subset of the input attributes, where these subset groupings capture the important dependencies in the modelling problem. A number of search strategies are introduced to find variable groupings based on both measures of correlation and improvement in discrimination. Learnt prototypes can then be used in conjunction with Semi-Naïve-Bayes and a defuzzification method to obtain estimated output values given inputs. In the experiments presented the Fuzzy Bayesian algorithm gives almost identical results to the ε-SVR and neural networks. However, the use of fuzzy labels provides flexible and transparent models that can be used with a high-level representation in terms of fuzzy labels to allow evaluation of queries expressed in natural language. This utilizes the calculus for appropriateness degree proposed in [9] and uses a similar system as evaluate expressions to that proposed for classification given in [13].

References

[1] A.J. Izenman, J.R. Wolf and Zürich Sunspot Relative Numbers, *The Mathematical Intelligencer*, Vol.7, No.1, 1985, pp 27–33.

[2] R.E. Bellman (1961), *Adaptive Signal Control Processing*, Princeton University Press, Princeton NJ.

[3] D. Dubois, F. Esteva, L. Godo, H. Prade, An Information-Based Discussion of Vagueness, *Fuzz-IEEE*, 2001.

[4] Irwin R. Goodman and Hung T. Nguyen (1985), *Uncertainty Models for Knowledge Based Systems - A Unified Approach Measurement of Uncertainty*, North-Holland - Amsterdam, New York, Oxford.

[5] S.R. Gunn, *Support Vector Machines for Classification and Regression*, University of Southampton, Faculty of Engineering and Applied Science, Department of Electronics and Computer Science, May 1998, `http://www.ecs.soton.ac.uk/~srg/publications/pdf/SVM.pdf`.

[6] S.R. Gunn, Matlab Support Vector Machine Toolkit, Version 2.1, `http://www.isis.ecs.soton.ac.uk/resources/svminfo`, May 2003.

[7] R. Hyndman and M. Akram, Time Series Data Library, `http://www-personal.buseco.monash.edu.au/~hyndman/TSDL/index.htm`, Monash University, Dept. of Econometrics and Business Statistics, Australia, 2003.

[8] I. Kononenko, Semi-Naïve Bayesian Classifier, *EWSL-91, Porto, Springer*, 1991, pp 206-219.

[9] J. Lawry, Label Semantics: A Formal Framework for Modelling With Words, *Symbolic and Quantitative Approaches to Reasoning With Uncertainty*, Lecture Notes in Artificial Intelligence, Springer-Verlag, 2001, pp 374–384.

[10] J. Lawry, Label Prototypes for Modelling with Words, *Proceedings of NAFIPS*, 2001.

[11] D.D. Lewis, Naïve (Bayes) at Forty: The Independence Assumption in Information Retrieval, *Machine Learning ECML-98*, LNAI 1398, 1998, pp 4-15.

[12] A. Papoulis (1965), *Probability, Random Variables, and Stochastic Processes*, McGraw-Hill, 1st edition.

[13] N.J. Randon and J. Lawry, Classification and Query Evaluation using Modelling with Words, To Appear: *Information Sciences Special Issue - Computing With Words: Models and Applications.*

[14] V.N. Vapnik (1995), *The Nature of Statistical Learning Theory*, Springer, New York.

[15] A.S. Weigend, B.A. Huberman, and D.E. Rumelhart, Predicting sunspots and exchange rates with connectionist networks. In M. Casdagli and S. Eubank, editors, *Non-linear Modelling and Forecasting, SFI Studies in the Sciences of Complexity, Proceedings*, Vol. XII, Addison-Wesley, 1992, pp 395–432

[16] L.A. Zadeh, Fuzzy Logic = Computing With Words, *IEEE Transaction on Fuzzy Systems*, Vol.4, No.2, 1996, pp 103–111.